Stanford Exploration Project, Report SERGEY, November 9, 2000, pages 555–??

Azimuth moveout: the operator parameterization and antialiasing

Sergey Fomel and Biondo L. Biondi[1]

ABSTRACT

A practical implementation of azimuth moveout (AMO) must be both computationally efficient and accurate. We achieve computational efficiency by parameterizing the AMO operator with the help of a transformed midpoint coordinate system. To achieve accuracy, the AMO operator needs to be carefully designed for antialiasing. We propose a modified version of Hale's antialiasing algorithm, which switches between interpolation in time and interpolation in space depending on the operator dips. The method is applicable to a vide variety of integral operators and compares favorably with the triangle filter technique. A simple synthetic example tests the applicability of the method to the AMO case.

INTRODUCTION

Azimuth moveout (AMO) was introduced by Biondi and Chemingui (1994a; 1994b) as an operator that transforms common-azimuth common offset seismic data from one vector offset to another. The time-and-space (Kirchhoff) formulation of AMO (Fomel and Biondi, 1995a,b) leads to a three-dimensional stacking operator, which includes four major components: the curvilinear surface of the summation path, the associated amplitude, the time filter, and the surface aperture (the range of integration). In this paper, we analyze two additional issues that are required for a successful practical implementation of the method: the operator parameterization and operator antialiasing.

The problem of parameterization arises because of the complicated time-dependent shape of the AMO aperture described in (Fomel and Biondi, 1995a). In this paper we show that the expressions for the summation path, the amplitudes, and the integration range have simple analytical forms when defined in the coordinate system of the input and output offset vectors.

The operator aliasing problem is common for a wide variety of integral (stacking) operators (Lumley et al., 1994). It is caused by the spatial undersampling of the summation path. When the integration path is parametrized in the spatial coordinate, as it is commonly done, the steeper part of the summation path becomes undersampled. The error introduced by the undersampling of the summation path is usually controlled by limiting the rate of change in the integrand (the input data) either by low-pass filtering (Gray, 1992), or by triangular filtering (Claerbout, 1992a). Unfortunately, in the case of AMO this simple methods are suboptimal

[1]**email:** sergey@sep.stanford.edu, biondo@sep.stanford.edu

because of the rapid changes in the summation path gradient that are encountered along the "ridges" of the AMO saddle. We therefore propose a new antialiasing method derived from the time-slice technique, which was developed by Dave Hale for DMO (Hale, 1991). Synthetic examples show the superiority of the new method compared with the triangle filtering.

PARAMETERIZATION

Azimuth moveout in the time-and-space domain is a three-dimensional integral operator (Fomel and Biondi, 1995a). Change (substitution) of the integrable variables as a method of integral simplification is well known in classic calculus. In the case of AMO, a convenient choice of the parameters of integration is of particular value because of the complicated shape of the operator aperture. In order to simplify the form of the AMO operator, thus reducing the cost of

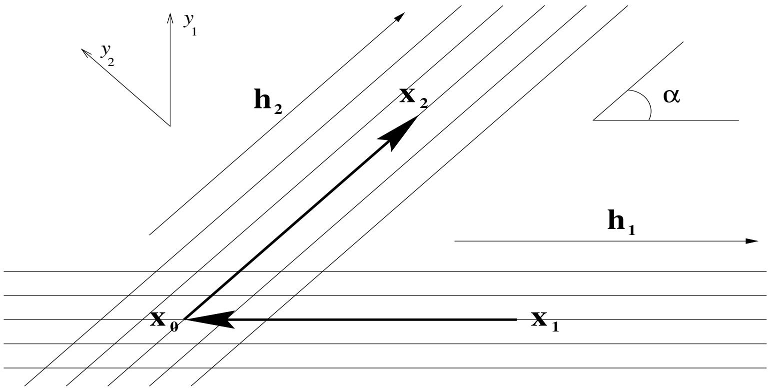

Figure 1: Schematic geometry of AMO and the transformed coordinate system. antial-amox12 [NR]

its computation, we propose the following substitution of variables. Let α_1 be the input offset azimuth, and α_2 be the output offset azimuth with respect to the midpoint coordinate system. Draw one axis (y_1) perpendicular to the direction of α_1, and the other axis (y_2) perpendicular to α_2. This defines a non-orthogonal coordinate system on the midpoint plane (Figure 1). The transformation of variables, written in the matrix form, is

$$\left[\begin{array}{c} y_1 \\ y_2 \end{array}\right] = \left[\begin{array}{cc} \cos\alpha_1 & -\sin\alpha_1 \\ \cos\alpha_2 & -\sin\alpha_2 \end{array}\right] \left[\begin{array}{c} x \\ y \end{array}\right] , \tag{1}$$

where x and y are the Cartesian coordinates of a midpoint in the original coordinate system. The Jacobian of transformation (1) is simply $|\sin\alpha_1 \cos\alpha_2 - \sin\alpha_2 \cos\alpha_1| = |\sin\alpha|$, where $\alpha = \alpha_2 - \alpha_1$ is the angle of azimuth rotation. Assuming that α is greater than zero, transformation (1) defines a spatially invariant rotational squeezing of the midpoint space. The special case of α equal to zero corresponds to the two-dimensional version of AMO, known as offset continuation (Bolondi et al., 1982; Chemingui and Biondi, 1994; Fomel, 1995), which can be handled separately. The expression for the traveltime of the AMO impulse response (formula

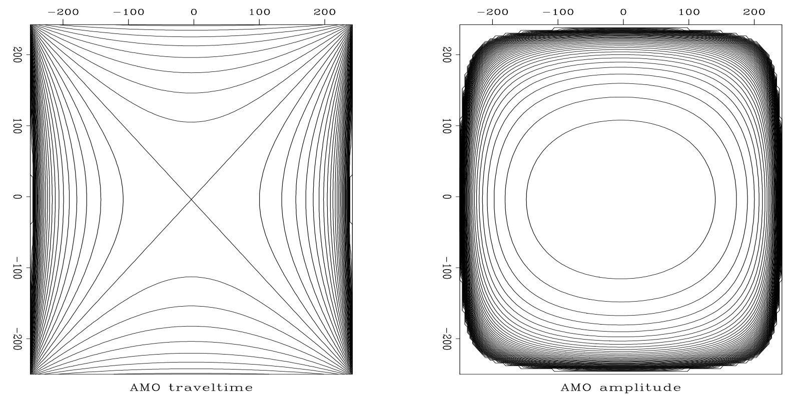

Figure 2: Traveltime and amplitude of the AMO impulse response in the transformed coordinate system. antial-amotta [ER]

(4) in (Biondi and Chemingui, 1994a)) transforms to

$$t_2 = t_1 \left|\frac{\mathbf{h_2}}{\mathbf{h_1}}\right| \sqrt{\frac{\mathbf{h_1^2} \sin^2\alpha - y_2^2}{\mathbf{h_2^2} \sin^2\alpha - y_1^2}}\,, \tag{2}$$

where $\{y_1, y_2\}$ is the midpoint separation in the transformed coordinate system. In the notation of Biondi and Chemingui, y_1 corresponds to $X \sin(\varphi - \theta_1)$, and y_2 corresponds to $X \sin(\varphi - \theta_2)$. One can see that the axes of the transformed coordinate system are now aligned along the axes of the AMO "saddle". The amplitude equations (Fomel and Biondi, 1995a; Chemingui and Biondi, 1995) are also simplified (Figure 2). What is more important, the transformation (1) affects the shape of the AMO aperture. The aperture limitation (21) from (Fomel and Biondi, 1995a) transforms after some heavy algebra to

$$\left(\frac{2}{v\,t_1}\right)^2 \leq \frac{\mathbf{h_1^2} \sin^2\alpha - y_2^2}{\mathbf{h_1^2} \sin^2\alpha} \left(\gamma_1^2 + \gamma_2^2 - 2\,\gamma_1\,\gamma_2 \cos\alpha\right)\,, \tag{3}$$

where

$$\gamma_1 = \frac{y_1}{\mathbf{h_2^2} \sin^2\alpha - y_1^2} = \frac{1}{t_2}\,\frac{\partial t_2}{\partial y_1}\,, \tag{4}$$

and

$$\gamma_2 = \frac{y_2}{\mathbf{h_1^2} \sin^2\alpha - y_2^2} = -\frac{1}{t_2}\,\frac{\partial t_2}{\partial y_2}. \tag{5}$$

The largest possible aperture corresponds to the zero input time (or zero velocity) and coincides with the interior of a rectangle centered at $\{\Delta y_1, \Delta y_2\} = \{0, 0\}$ with the sides of the rectangle equal to $2\,|\mathbf{h_2}|\,\sin\alpha$ and $2\,|\mathbf{h_1}|\,\sin\alpha$. With the time increase, the aperture gradually decreases in size, and its shape approaches a quasi-elliptical form (Figure 3). From the computational point of view, it is convenient to evaluate the right-hand side of inequality (3) outside of the input time loop and use this inequality to limit the range of times for each point of the operator.

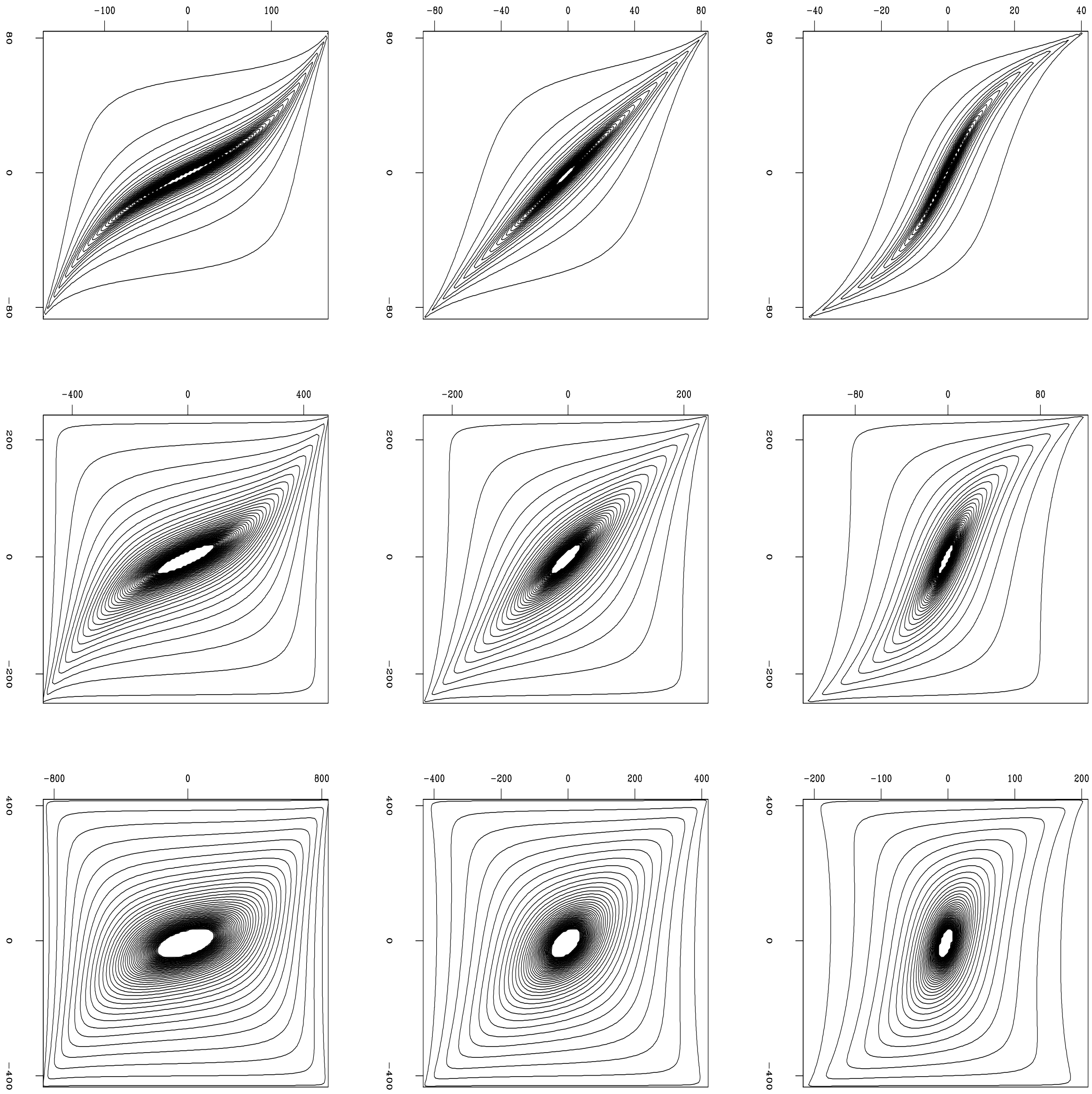

Figure 3: AMO aperture in the transformed coordinate system as a function of the input time. The different plots correspond to different geometries of AMO. From top to bottom: the angle of azimuth rotation α changes from 10 degrees (top) to 30 degrees (middle) and 60 degrees (bottom). From left to right: the ratio of offsets $|\mathbf{h_2}/\mathbf{h_1}|$ changes from 1/2 (left) to 1 (middle) and 2 (right). antial-amoapp [ER]

ANTIALIASING

The operator aliasing problem, as opposed to data aliasing and image aliasing, is discussed in detail by Lumley et al. (1994). It arises when the slope of the operator traveltime exceeds the limit, defined by the time and space sampling of the data (the Nyquist frequencies) (Claerbout, 1992a). Even if the input data are not aliased, operator aliasing can cause severe distortions in the output. Several successful techniques have been proposed in the literature to overcome the operator aliasing problem. SEP' s favorite invention is local triangle filtering (Claerbout, 1992a; Bevc and Claerbout, 1992, 1993; Lumley et al., 1994; Bevc and Lumley, 1994), which has been extensively tested on 2-D and 3-D migration, DMO (Blondel, 1993), and wave-equation datuming (Bevc, 1992). A different approach to antialiasing is suggested by Hale (1991) for the integral dip moveout. In this paper, we reformulate the main principle of Hale's approach to design an efficient antialiasing technique, alternative to triangle filtering.

Triangle filters

The idea of the triangle filtering (Claerbout, 1992a; Lumley et al., 1994) follows from the well-known Nyquist sampling criterion, applied on the stacking-type operator:

$$\Delta x \leq \frac{\Delta t}{|\partial t / \partial x|} , \tag{6}$$

where $t(x)$ is the traveltime of the operator imuplse response (or the summation path of the its adjoint). In the steep parts of the traveltime curve, the sampling criterion (6) is not satisfied, which causes aliasing artifacts in the output data. To overcome this problem, the method of triangle filtering suggests convolving the traces of the generated impulse response with a triangle-shaped filter of the length

$$\delta t = \Delta x \, |\partial t / \partial x| \, . \tag{7}$$

Cascading operators of causal and anticausal numerical integration is an efficient way to construct the desired filter shape (Bevc and Claerbout, 1993). Triangle filters approximate the ideal (sinc) low-pass time filters. The idea behind low-pass filtering as a tool of antialiasing (Gray, 1992) is illustrated in Figure 4. When a steeply dipping event is included in the operator, its counterpart in the frequency domain wraps around to produce the aliasing artifacts. Those are removed by a dip-dependent low-pass filtering. The method of triangle filtering is less evident in the case of a three-dimensional integral operator. We can take the length of a triangle filter proportional to the absolute value of the time gradient (Lumley, 1993), the maximum of the gradient components in the two directions of the operator space, or the sum of these components. The latter follows from considering the 3-D operator as a double integration in space. Decoupling the 3-D integral into a cascade of two 2-D operators suggests convolving two triangle filters designed with respect to two coordinates of the operator. In this case, the length of the resultant filter is approximately equal to

$$\delta t = \Delta x \, |\partial t / \partial x| + \Delta y \, |\partial t / \partial y| \, , \tag{8}$$

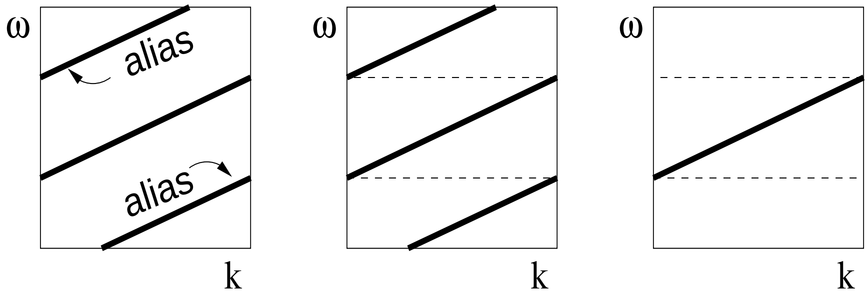

Figure 4: Schematic illustration of low-pass antialiasing (triangle filters). The aliased events are removed by low-pass filtration on the temporal frequency axis. The width of the low-pass filter depends on dips of the aliased events. antial-amolow [NR]

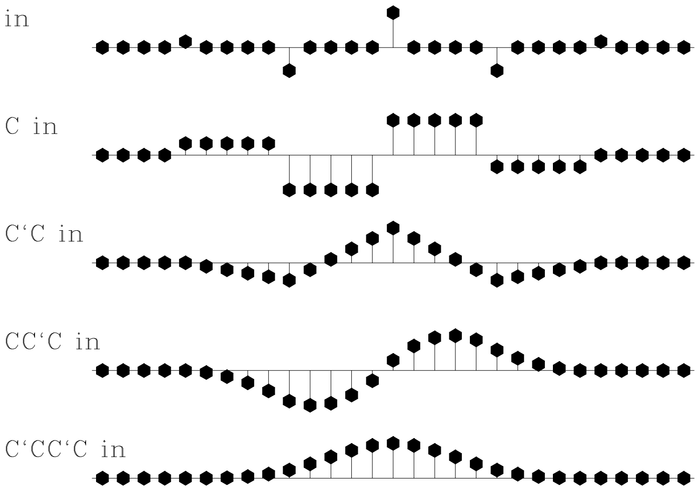

Figure 5: Building the smoothed filter for 3-D antialiasing by successive integration of a five-point wavelet. C denotes the operator of causal integration, C' denotes its adjoint (the anticausal integration). The result is equivalent to the convolution of two equal triangle filters. antial-amoflt [ER]

and its shape is smoother than that of a triangle filter (Figure 5). In the case of azimuth moveout, the width of the antialising filter is derived from formula (8) and the travel-time equation (2) as

$$\delta t_2 = \Delta y_1 \, |\partial t/\partial y_1| + \Delta y_2 \, |\partial t/\partial y_2| = t_2(|\gamma_1| \, \Delta y_1 + |\gamma_2| \, \Delta y_2) \,. \tag{9}$$

The triangle filtering method proven an efficient tool in the design of stacking operators of different types. However, we see the following two disadvantages of applying it to the AMO case:

1. The saddle shape of the AMO operator introduces rapid changes in the length and direction of the traveltime gradient. It leads to an inexact estimation of the triangle length at the curved parts of the operator. Consequently, the high-frequency part of the output can be distorted, causing a loss in the image resolution.

2. For large input times, most of the energy of the AMO operator is concentrated in the flat part of its traveltime surface (the middle part of the "saddle"). This part does not contain aliased energy and does not require any sophistication in the time interpolation.

Hale's method

Considering the case of integral DMO, Hale (1991) points out that the steep parts of the operator, while aliased in the space (midpoint) coordinate, are not aliased with respect to the time coordinate. He suggests replacing the conventional $t(x)$ parameterization of the DMO impulse response by $x(t)$ parameterization. Conventionally, the integral operators are implemented by shifting the input traces in space and transforming them in time. According to Hale's method, the traces are shifted in time and transformed along the $x(t)$ trajectories in space. Interpolation in time, required in the conventional approach, is replaced by interpolation in space. The idea of Hale's method is related to the idea of the "pixel-precise velocity transform", introduced by Claerbout (1990; 1992b). The steep parts of the operator satisfy the criterion

$$\Delta t \leq \frac{\Delta x}{|\partial x/\partial t|} \,, \tag{10}$$

which is the the obvious reverse of inequality (6). Therefore, they are not aliased if defined on the time grid. In these parts one can perform the operator by constant time shifts equal to the time sampling interval Δt. In the parts where the criterion (10) is not valid (the flat part of the DMO operator), Hale suggests reducing the length of the time shifts according to equality (7), where δt becomes less than Δt. He formulates the following principle of operator antialiasing:

> *To eliminate spatial aliasing, simply never allow successive time shifts applied to the input trace to differ by more than one time sampling interval. Further restrict the difference between time shifts so that the spacing between the corresponding output trajectories never exceeds the CMP sampling interval*

We illustrate the idea of Hale's method in Figure 6. Increasing the density of spatial sampling by small successive time shifts implies increasing the Nyquist boundaries of the spatial spectrum (wavenumber). Further interpolation is a low-pass spatial filter removing the parts of the spectrum beyond the Nyquist frequency of the output. If the dip of the operator does not vary between neighboring traces (the operator is a straight line as in the slant stack case), Hale's approach produce essentially the same result as low-pass filtering. Triangle filters in this case approximately correspond to linear interpolation in space between adjacent traces (Nichols, 1993). The difference between the two approaches occurs if the local dip varies in space (the case of a curved operator, such as DMO). In this case, Hale's approach provides a more accurate space interpolation of the operator and preserves the high-frequency part of its spectrum from distortion. Hale's method has proven to preserve the amplitude of flat reflectors from

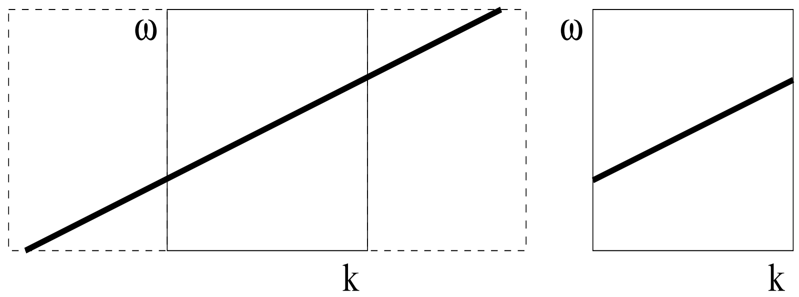

Figure 6: Schematic illustration of Hale's antialiasing. The aliased events are removed by spatial interpolation. In the frequency domain, the interpolation consists of widening and low-passing on the wavenumber axis. The low-pass spatial filtering does not depend on dip. antial-amosft [NR]

aliasing distortions, which is the simplest antialiasing test on a DMO operator. We see the most valuable advantage of this method in the fact that the implied low-pass spatial filtering (interpolation) does not depend on the operator dip and is controlled by the Nyquist boundary of the spectrum only (compare Figures 4 and 6). This is especially important, when the local dip of the operator changes rapidly and therefore cannot be estimated precisely by finite-difference approximation at spatially separated traces. Such a situation is common in DMO and AMO integral operators, as well as in prestack Kirchhoff migration. A weakness of the method is the necessity to switch from interpolation in space to two-dimensional interpolation in both the time and the space variables, when trying to construct the flat part of the operator. In the case of AMO, the 2-D spatial interpolation arises as a result of building the operator in the transformed coordinate system. However, we would prefer to avoid the expense of the additional time interpolation required by Hale's method of antialiasing.

Proposed technique

We use the reciprocity of the time parameterization and the space parameterization of integral operators, discovered by Hale, to develop the following antialiasing technique. For simplicity, let us consider the two-dimensional case first. The linearity of a two-dimensional integral operator allows us to decompose this operator into two terms. The first term corresponds to the steep part of the travel-time function, satisfying the time-sampling criterion (10). The second term corresponds to the flat part of the traveltime, which satisfies the midpoint-sampling criterion (6). The first part is not aliased with respect to the time sampling interval, while the second one is not aliased with respect to the space sampling. We apply a simple linear interpolation in time to construct the flat part. Reciprocally, linear interpolation in space is applied to construct the steep part of the operator in the fashion of Hale's time-shifting method. Linear interpolation in this case is a cheap substitution for the errorless, but computationally expensive sinc interpolation. The amplitude difference between the two integrals is simply the Jacobian term

$$\frac{\mathrm{amp}_t}{\mathrm{amp}_x} = \left|\frac{\partial x}{\partial t}\right| \frac{\Delta t}{\Delta x} = \frac{\Delta t}{\delta t} \leq 1 \; . \tag{11}$$

According to the proposed modification, Hale's antialiasing principle is reformulated, as follows:

> *In the steep part of an integral operator, never allow successive time shifts applied to the input trace to differ by more than one time sampling interval. In the flat part of the operator, never allow successive space shifts to differ by more than one space sampling interval.*

Figure 7, borrowed from **Basic Earth Imaging** (Claerbout, 1995), illustrates the basic idea of the proposed technique. It clearly shows the difference between the flat and steep parts of migration hyperbolas. To view the reciprocity, rotate the figure by 90 degrees. The reader

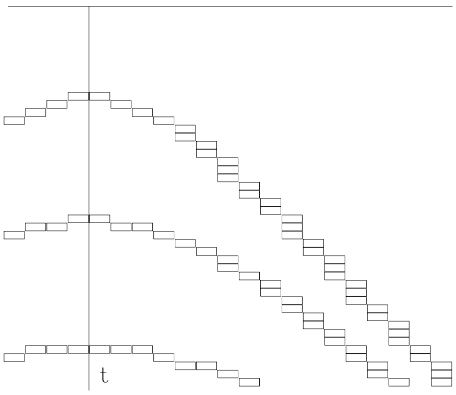

Figure 7: Figure borrowed from **BEI** to illustrate the reciprocity antialiasing. The flat parts of the hyperbolas require interpolation in time. The steep parts of the hyperbolas require interpolation in space. antial-amotra [ER]

familiar with Ratfor can examine the details of the algorithm in the post-stack migration program, listed in Appendix. The program is based on the tutorial `kirchfast` program in **BEI**. The nearest neighbor interpolation is replaced by linear interpolation, and the two parts of the program stand for the steep-dip and low-dip parts of the operator. The program was not optimized for a better performance. To compare the proposed antialiasing with triangle filtering, we test the antialiased migration program on SEP's canonical 2-D synthetic tests. Figure 8 shows a simple model and the modeling results from aliased (the nearest neighbor interpolation) modeling, triangle antialiasing and the proposed reciprocity method. The modeling results were migrated with the corresponding migration operators to obtain the image of the model in Figure 9. Both the triangle filtering and the proposed method succeeded in removing the major aliasing artifacts. However, the reciprocity method demonstrates a higher resolution and a better preservation of the frequency content. These properties are examined more

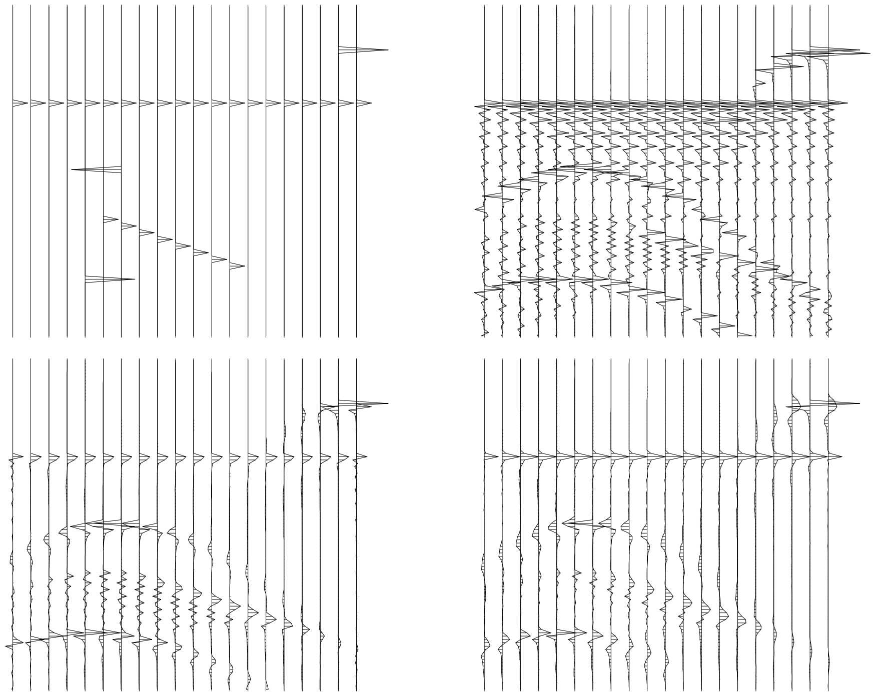

Figure 8: Top left is a synthetic model. Top right is modeling without antialiasing. Bottom left is modeling with reciprocity antialiasing (the proposed method). Bottom right is modeling with triangle filter antialiasing. antial-amomod [ER]

closely in the next synthetic example. Figure 10 shows a more sophisticated model that contains a fault, an uncomformity and faulting structures (Claerbout, 1995). For better displaying, we extract the central part of the model and compare it with the migration results of different methods in Figure 11. Comparing the plots shows that the reciprocity method successfully removes the aliasing artifacts (round-off errors) of the aliased (nearest neighbor interpolation) migration. At the same time, it is less harmful to the high-frequency components of the data than triangle filtering. This conclusion finds an additional support in Figure 12 that displays the average spectrum of the image traces for different methods. Both of the antialiasing methods remove the high-frequency artifacts of the nearest neighbor modeling and migration. The reciprocity method performs it in a gentler way, preserving the high-frequency components of the model. The algorithm sequence of the antialiased migration is illustrated in Figures 13 and 14. The two plots in Figure 13 show the steep-dip and flat-dip modeling respectively. The superposition of these two terms is the resultant antialiased data shown in the left plot of Figure 15. The right plot of Figure 15 shows the migrated image obtained by adding the flat-dip (left of Figure 14) and steep-dip (right of Figure 14) migrations. We have compared the performance of the antialiased migration with that of the aliased migration and the migration with triangle filtering. The test data set included 500 by 250 data points with $\Delta t = 0.004$ sec, and $\Delta x = 25$ m. The CPU time of different routines on the HP 9000-735/99 workstation is charted in Figure 16. The figure shows that the performance of the reciprocity antialiasing increases

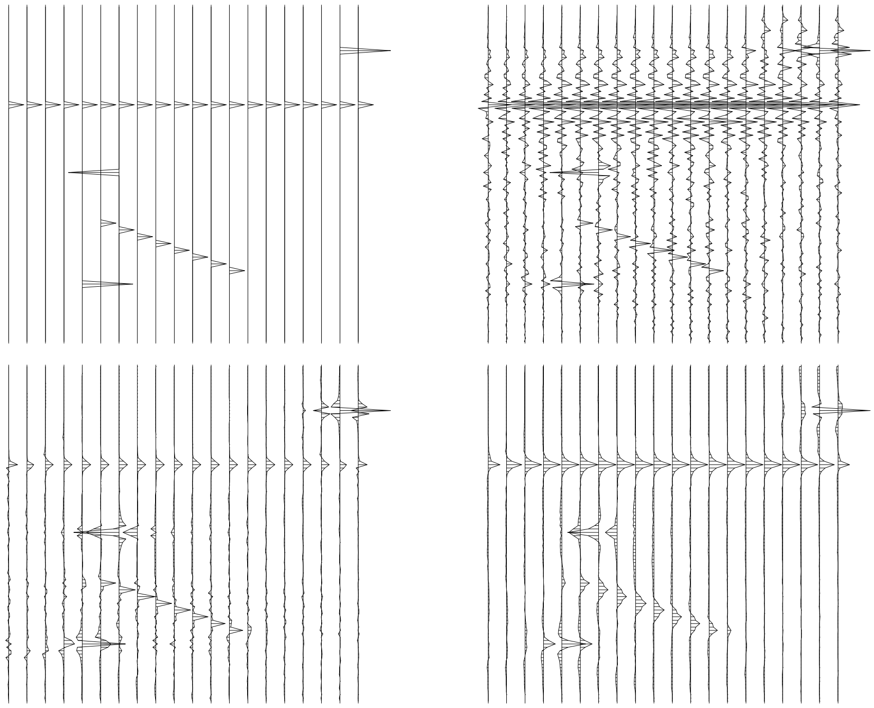

Figure 9: Top left plot is the synthetic model. The other plots are migrations of the corresponding data shown in the previous figure . Top right is a migration without antialiasing. Bottom left is a migration with reciprocity antialiasing (the proposed method). Bottom right is a migration with triangle filter antialiasing. antial-amomig [ER]

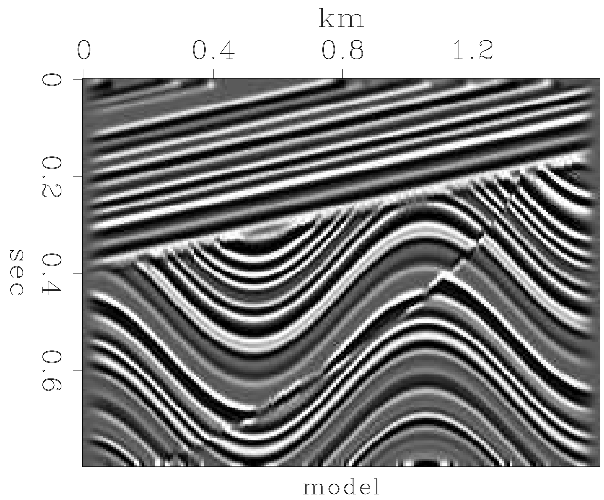

Figure 10: Synthetic model used to test the antialiased migration program. antial-amosmo [ER]

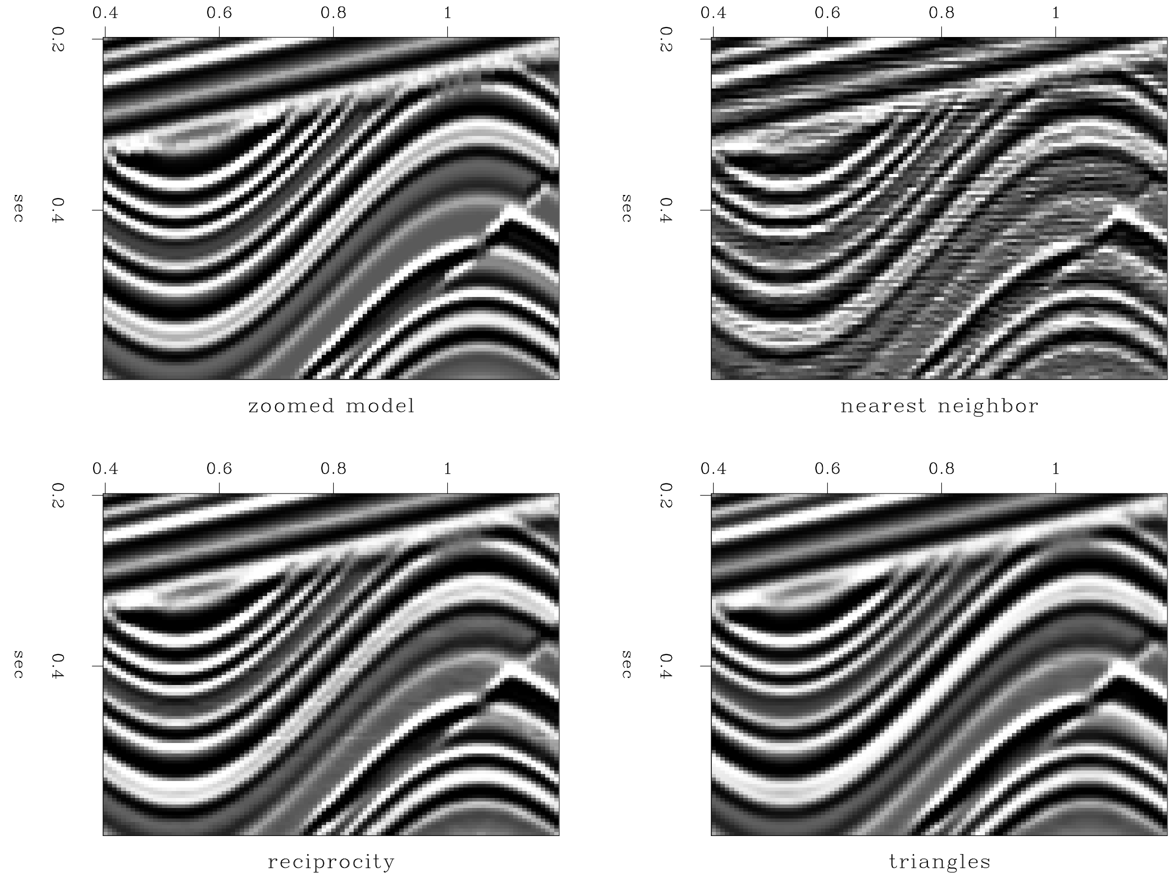

Figure 11: Top left plot is a zoomed portion of the synthetic model. The other plots are migrated images. Top right is a migration without antialiasing. Bottom left is a migration with reciprocity antialiasing (the proposed method). Bottom right is a migration with triangle filter antialiasing. antial-amosmi [ER]

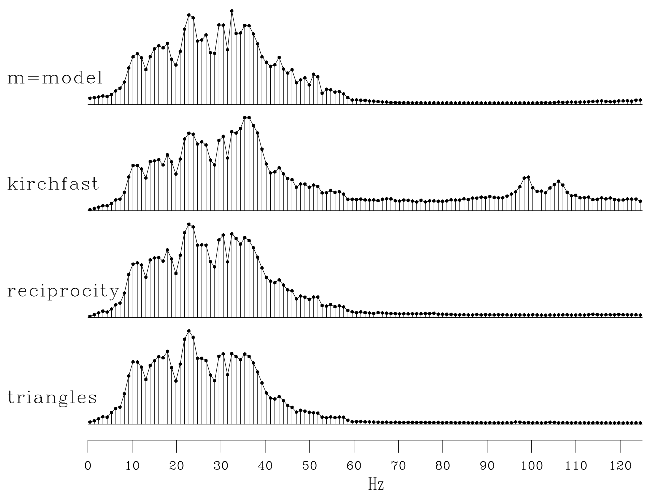

Figure 12: Top is the spectrum of the model. The other plots are the spectra of the migrated images. The second plot corresponds to the modeling/migration without account for antialiasing. The third plot is modeling/migration with the reciprocity antialiasing. The bottom plot is modeling/migration with triangle antialiasing. antial-amospe [ER]

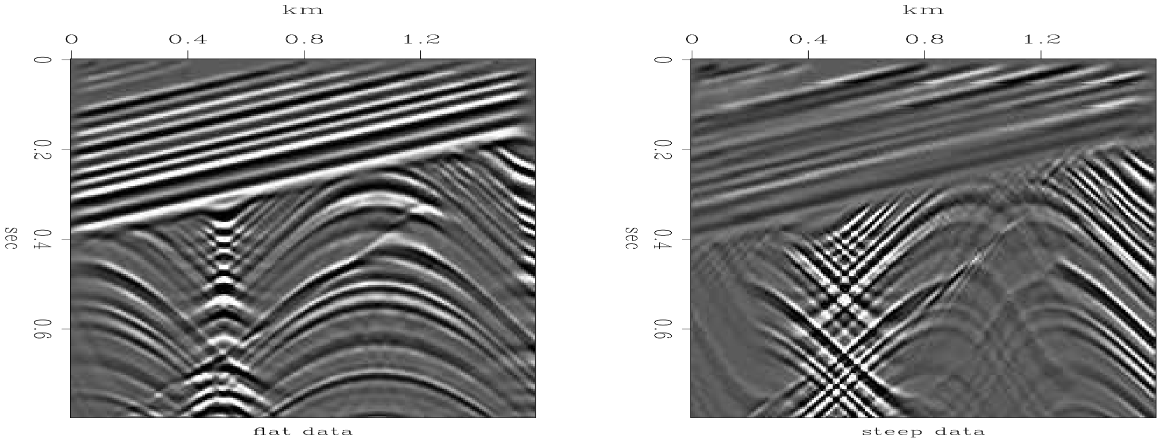

Figure 13: Antialiased modeling. Left corresponds to the flat-dip term. Right corresponds to the steep-dip term. antial-amormo [ER]

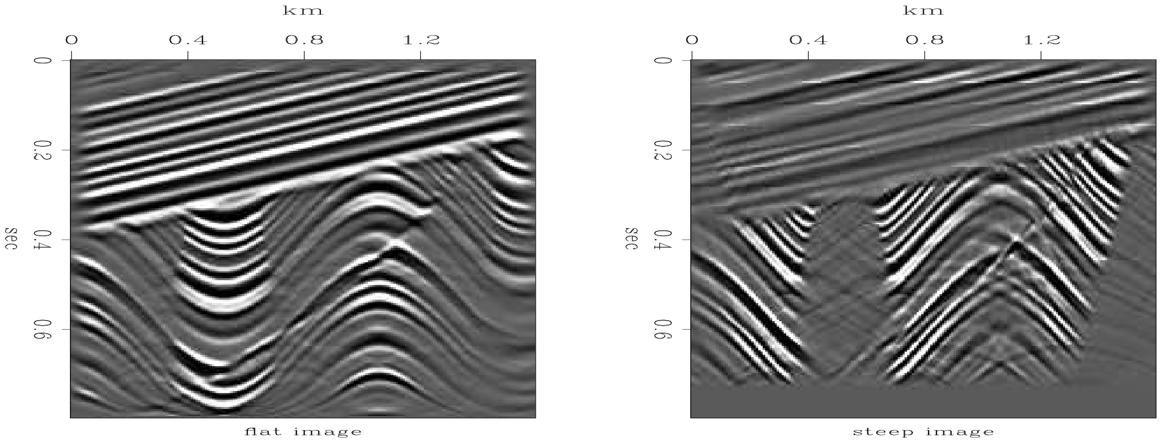

Figure 14: Antialiased migration. Left corresponds to the flat-dip term. Right corresponds to the steep-dip term. antial-amormi [ER]

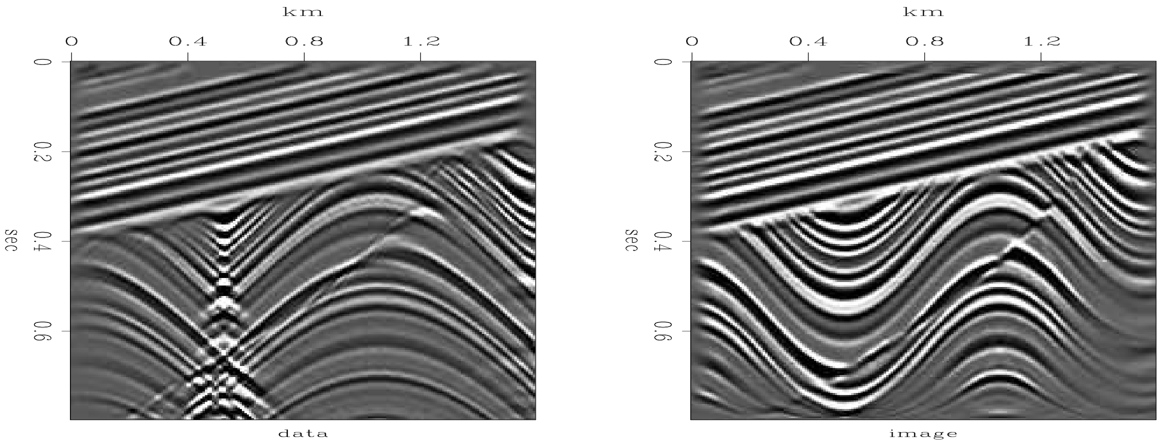

Figure 15: Antialiased modeling and migration. Left is the superposition of the flat-dip and steep-dip modeling. Right is superposition of the flat-dip and steep-dip migration. antial-amormm [ER]

with increase of the migration velocity. This surprising behavior is explained by the fact that high-velocity migration hyperbolas require a smaller number of expensive computations in the steep (aliased) parts. It allows us to expect a high performance of the method in application to the curvilinear operators with limited aperture (DMO, offset continuation, AMO). In the test employed, the overall performance of our migration program appeared higher than that of the `kaafast` program (Bevc and Claerbout, 1992). The proposed method of antialiasing is easily

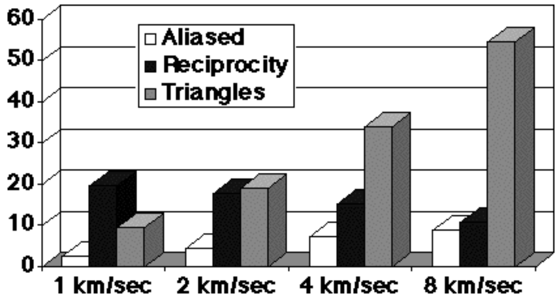

Figure 16: CPU time of migration programs on HP 9000-735 versus the constant migration velocity used in the experiment. antial-amochp [NR]

generalized to the case of a three-dimensional integral operator, such as azimuth moveout. In this case, one needs to consider three different parameterizations: $t(x,y)$, $x(t,y)$, and $y(t,x)$ and switch from one of them to another according to the rule:

- if $\Delta t \geq \Delta x\,|\partial t/\partial x|$ and $\Delta t \geq \Delta y\,|\partial t/\partial y|$, use $t(x,y)$,
- if $\Delta x \geq \Delta t\,|\partial x/\partial t|$ and $\Delta x \geq \Delta y\,|\partial x/\partial y|$, use $x(t,y)$,
- if $\Delta y \geq \Delta t\,|\partial y/\partial t|$ and $\Delta y \geq \Delta x\,|\partial y/\partial x|$, use $y(t,x)$.

AMO TEST

Our first synthetic test of the AMO operator is a simple diffraction in a constant velocity medium. We modeled a common-azimuth data set over a diffraction point with an offset of 500 meters and a regular midpoint grid 20 by 20 meters. The AMO operator was designed to rotate the offset azimuth of the data by 30 degrees. The results are compared with the modeled data in Figure 17. Azimuth moveout has succeeded in reconstructing the true geometry of the desired output, though it did not behave perfectly with respect to the amplitudes and boundary effects. The corresponding AMO impulse response is shown in crossline and inline sections in Figure 18. For simplicity, this impulse response doesn't include the derivative filter required for the complete definition of AMO. Figure 19 illustrates the antialising applied to AMO. The "top" view in the time direction shows how the antialiased AMO operator is constructed from the flat-dip and steep-dip parts.

CONCLUSIONS

On the way from the AMO theory to practice, we have solved two important problems, crucial for the successful implementation of the method.

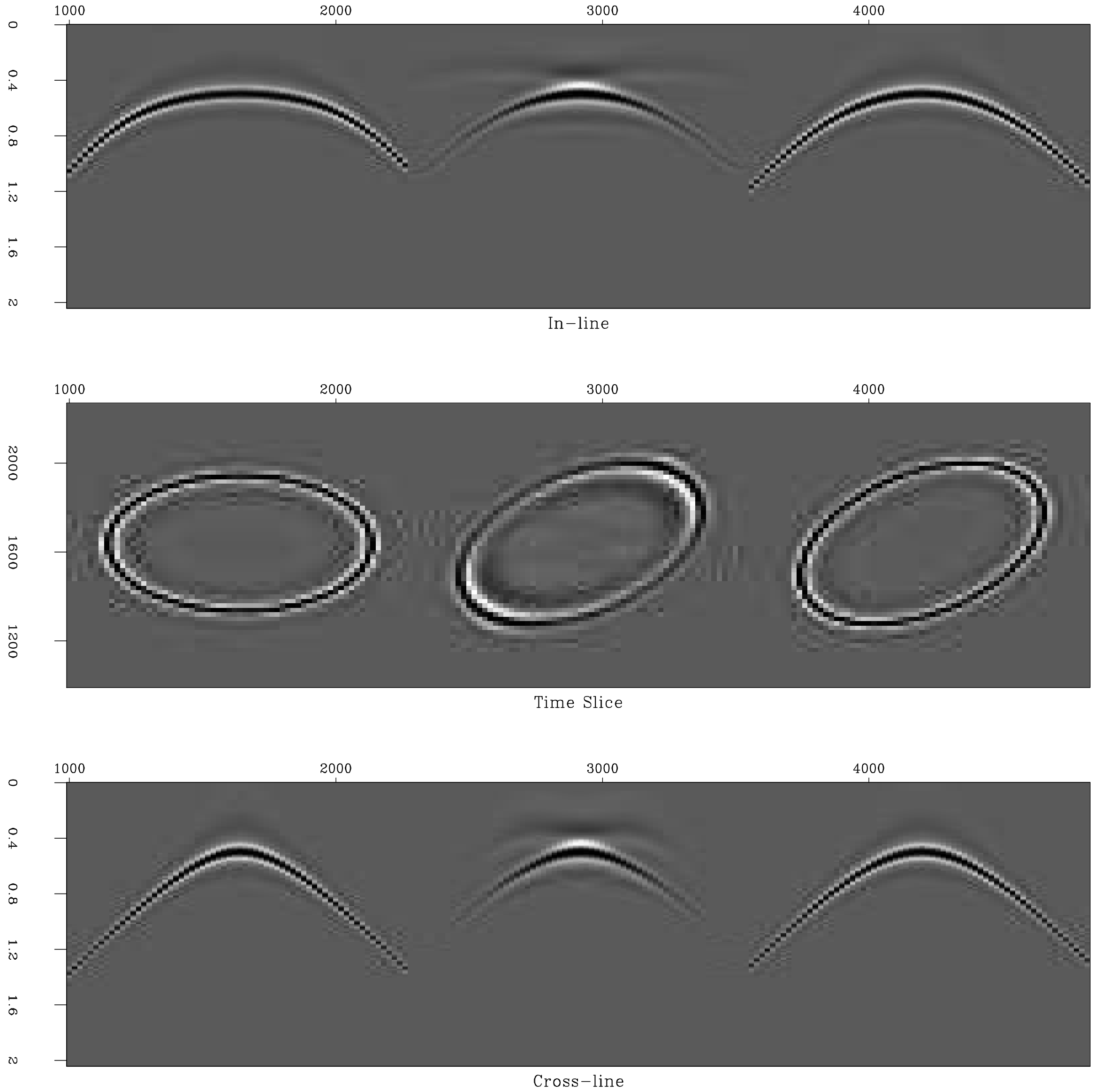

Figure 17: Diffraction test on azimuth moveout. Left is the input, right is the desired output, middle is the output of AMO. antial-amoimp [CR]

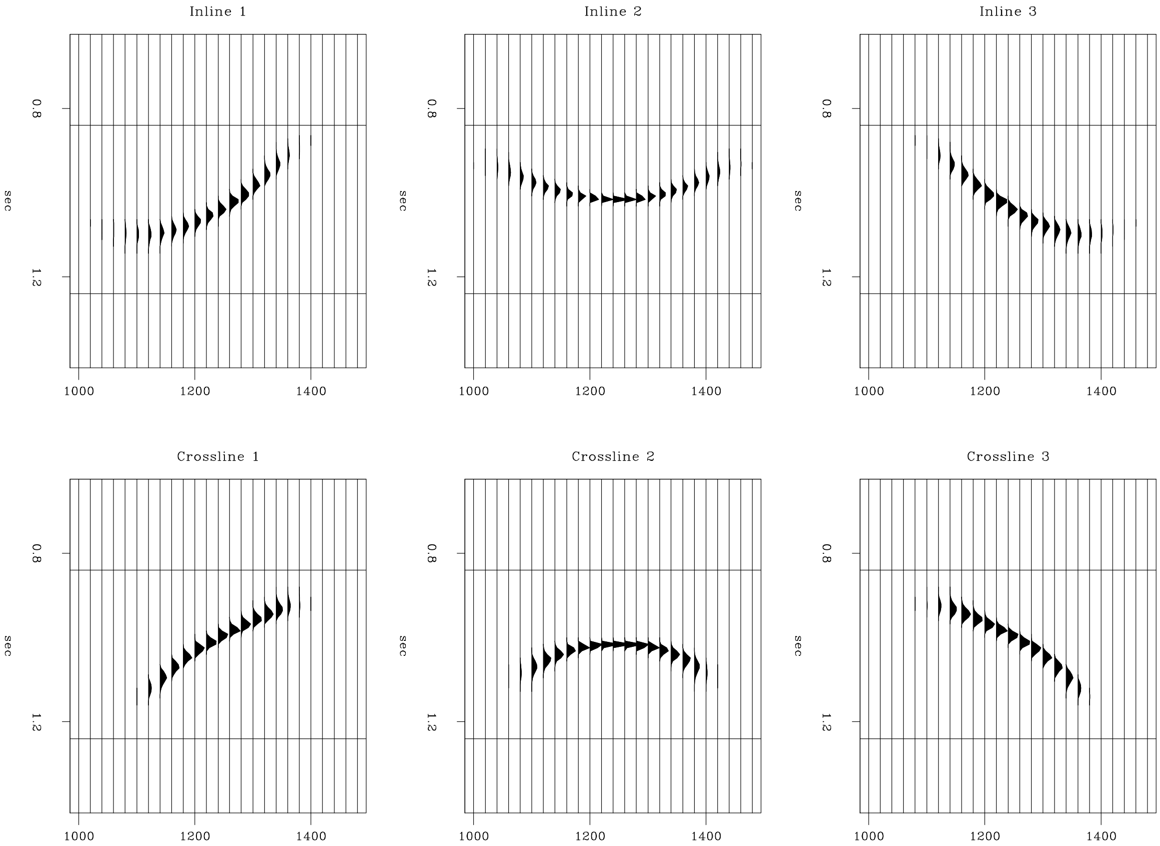

Figure 18: AMO impulse response in crossline (bottom) and inline (top) sections. The AMO geometry corresponds to h_1=500 meters, h_2=500 meters, α_1=0, and α_2=30 degrees. The derivative filter is not included. antial-amoimr [ER]

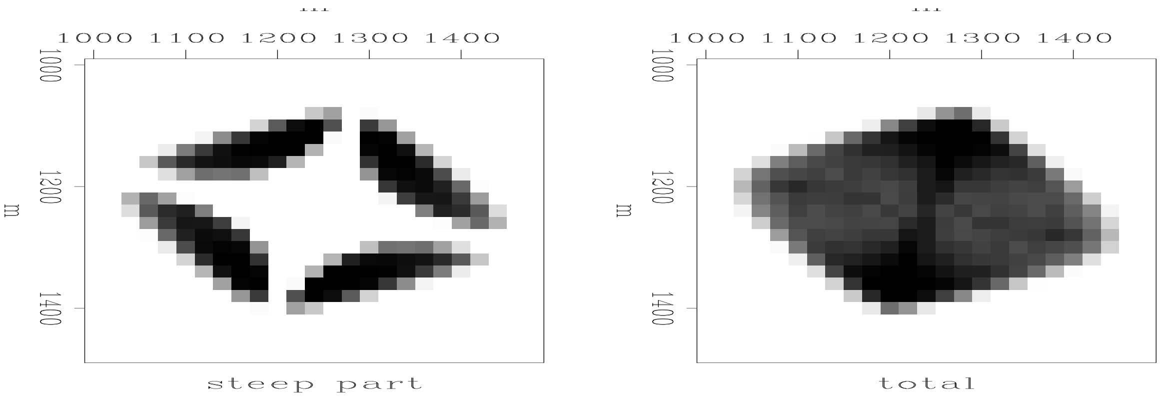

Figure 19: "Top" view on the antialiased AMO operator (stacked time slices.) The AMO impulse response is created by superposition of the flat-dip and steep-dip parts. antial-amocon [ER]

1. We have shown that a convenient parameterization of the AMO operator enables fast and accurate computation of the operator components, including the spatial aperture.
2. We have introduced a new method of antialising integral operators, modified from Hale's approach to antialised DMO. The method compares favorably with the triangle filtering technique. Its main advantage is in preserving the high-frequency part of the data spectrum, which leads to a better resolution. It also allows for an easy control of the amplitudes and possesses a sufficient numerical efficiency.

The parameterization and antialising have been applied to enhance the AMO operator with respect to both accuracy and computational efficiency. Currently we are in a process of testing the integral antialised AMO on synthetics and wait for real data tests to arrive.

REFERENCES

Bevc, D., and Claerbout, J., 1992, Fast anti-aliased Kirchhoff migration and modeling: SEP–**75**, 91–96.

Bevc, D., and Claerbout, J., 1993, Choice of integration method for anti-aliased Kirchhoff migration: SEP–**77**, 295–302.

Bevc, D., and Lumley, D. E., 1994, When is anti-aliasing needed in Kirchhoff migration?: SEP–**80**, 467–476.

Bevc, D., 1992, Kirchhoff wave-equation datuming with irregular acquisition topography: SEP–**75**, 137–156.

Biondi, B., and Chemingui, N., 1994a, Transformation of 3-D prestack data by Azimuth Moveout: SEP–**80**, 125–143.

Biondi, B., and Chemingui, N., 1994b, Transformation of 3-D prestack data by azimuth moveout (AMO): 64th Ann. Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1541–1544.

Blondel, P., 1993, Constant-velocity anti-aliasing three-dimensional integral dip moveout: SEP–**77**, 49–58.

Bolondi, G., Loinger, E., and Rocca, F., 1982, Offset continuation of seismic sections: Geophys. Prosp., **30**, no. 6, 813–828.

Chemingui, N., and Biondi, B., 1994, Coherent partial stacking by offset continuation of 2-D prestack data: SEP–**82**, 117–126.

Chemingui, N., and Biondi, B., 1995, Amplitude preserving AMO from true amplitude DMO and inverse DMO: SEP–**84**, 153–168.

Claerbout, J. F., 1990, Hyperbola tricks: SEP–**65**, 241–246.

Claerbout, J. F., 1992a, Anti aliasing: SEP–**73**, 371–390.

Claerbout, J. F., 1992b, Earth Soundings Analysis: Processing Versus Inversion: Blackwell Scientific Publications.

Claerbout, J. F., 1995, Basic earth imaging: SEP.

Fomel, S., and Biondi, B., 1995a, The time and space formulation of azimuth moveout: SEP–**82**, 25–37.

Fomel, S., and Biondi, B. L., 1995b, The time and space formulation of azimuth moveout: 65th Ann. Internat. Meeting, Soc. Expl. Geophys., Expanded Abstracts, 1449–1452.

Fomel, S., 1995, Amplitude preserving offset continuation in theory. Part 1: The offset continuation equation: SEP–**84**, 179–196.

Gray, S. H., 1992, Frequency-selective design of the Kirchhoff migration operator: Geophysical prospecting, **40**, 565–571.

Hale, D., 1991, A nonaliased integral method for dip moveout: Geophysics, **56**, no. 6, 795–805.

Lumley, D. E., Claerbout, J. F., and Bevc, D., 1994, Anti-aliased Kirchhoff 3-D migration: SEP–**80**, 447–490.

Lumley, D. E., 1993, Anti-aliased Kirchhoff 3-D migration: A salt intrusion example: SEP–**77**, 1–18.

Nichols, D., 1993, Integration along a line in a sampled space: SEP–**77**, 283–294.

APPENDIX A

POST-STACK TIME MIGRATION (FAST, ANTIALIASED)

```
module kirchnew {
  integer                      :: nt, nx, sw
  real                         :: t0, dt, dx
  real, dimension (:), pointer :: vrms
#% _init (vrms, t0,dt,dx, nt,nx, sw)
#% _lop (modl(nt,nx), data(nt,nx))
  integer :: ix,iz,it,ib,iy, minx(2),maxx(2), is,i
  real  :: amp,t,z,b,db,f,g
        maxx(1) = nx;   minx(2) = 1
  do iz= 1,nt-1 {     z = t0 + dt * (iz-1)              # vertical traveltime
  do it= nt,iz+1,-1 { t = t0 + dt * (it-1)              # time shift
                  b = sqrt(t*t - z*z); db = dx*b*2./(vrms(iz)*t)
                                        if(db < dt .or. sw == 1) exit
                  f = 0.5*vrms(iz)*b/dx; iy = f; f = f-iy; i = iy+1; g = 1.-f
                                        if(i >= nx)             cycle
        amp = (z / (t+dt)) * sqrt(nt*dt / (t+dt)) * (dt / db)

        minx(1) = 1+i;  maxx(2) = nx-i
  do is= 1,2   {  iy = -iy; i = -i              # two branches of hyperbola
  do ix=        minx(is),       maxx(is) {
                if( adj)
                        modl(iz,ix)    = modl(iz,ix) + data(it,ix+iy)*amp*g +
                                                       data(it,ix+i )*amp*f
                else {
                        data(it,ix+iy) = data(it,ix+iy) + modl(iz,ix)*amp*g
                        data(it,ix+i ) = data(it,ix+i ) + modl(iz,ix)*amp*f
                }
        }}}
  do ib= 0, nx-1 {         b = dx*ib*2./vrms(iz); iy = ib       # space shift
                 t = sqrt(z*z  + b*b); db = dx*b*2./(vrms(iz)*t)
                                        if(db > dt .or. sw == 2) exit

                 f = (t-t0)/dt; i = f; it = i+1; f = f-i ; i = it+1; g = 1.-f
                                        if( i > nt)             exit

        amp = (z / (t+dt)) * sqrt(nt*dt / (t+dt)); if(ib == 0) amp = amp*0.5

        minx(1) = 1+iy; maxx(2) = nx-iy
  do is= 1,2   {        iy = -iy                # two branches of hyperbola
  do ix=        minx(is),       maxx(is) {
                if( adj)
                        modl(iz,ix)    = modl(iz,ix) + data(it,ix+iy)*amp*g +
                                                       data(i ,ix+iy)*amp*f
                else {
                        data(it,ix+iy) = data(it,ix+iy) + modl(iz,ix)*amp*g
                        data(i ,ix+iy) = data(i ,ix+iy) + modl(iz,ix)*amp*f
                }
        }}}
  }
}
```